

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

**Detekce dopravních značek v obrazech
pomocí obrazových příznaků**

**Traffic Signs Detection in Images Using
Image Features**

Zadání bakalářské práce

Student:

Stanislav Sehnal

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Detekce dopravních značek v obrazech pomocí obrazových příznaků
Traffic Signs Detection in Images Using Image Features

Jazyk vypracování:

čeština

Zásady pro vypracování:

Detekce dopravních značek za pomoci obrazů může být využita zejména v automobilech, které jsou vybaveny kamerami. Například pro zvýšení bezpečnosti v silničním provozu nebo pro navigování samoříditelných vozidel.

1. Seznamte se se základními pojmy v oblasti detekce a rozpoznání objektů v obrazech.
2. Seznamte se s klasifikátory, které mohou být v tomto problému aplikovány (například SVM).
3. Seznamte se s obrazovými příznaky, které mohou být v tomto problému aplikovány (například HOG, Haar, LBP).
4. Implementujte detektor dopravních značek.
5. Využít můžete například knihovny OpenCV nebo Dlib.
6. Experimentálně ověřte funkčnost, přesnost a rychlost detektoru.
7. Své závěry řádně zdokumentujte v textu práce.

Seznam doporučené odborné literatury:

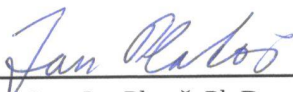
- [1] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li and S. Hu, "Traffic-Sign Detection and Classification in the Wild," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 2110-2118.
- [2] G. Overett and L. Petersson, "Large scale sign detection using HOG feature variants," 2011 IEEE Intelligent Vehicles Symposium (IV), Baden-Baden, 2011, pp. 326-331.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Radovan Fusek, Ph.D.**

Datum zadání: 01.09.2018

Datum odevzdání: 30.04.2019


doc. Ing. Jan Platoš, Ph.D.
vedoucí katedry




prof. Ing. Pavel Brandštetter, CSc.
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární
prameny a publikace, ze kterých jsem čerpal.

V Ostravě 29. dubna 2019

.....
Sehnal

Rád bych poděkoval Ing. Radovanovi Fuskovi, Ph.D. za odbornou pomoc a konzultaci při vytváření této bakalářské práce.

Abstrakt

Cílem této bakalářské práce byla detekce dopravních značek za pomoci obrazů v automobilech, které jsou vybavené kamerami. Pomocí tohoto systému můžeme zvýšit bezpečnost v silničním provozu a rovněž lze tento systém využít k navigaci samoříditelných vozidel. Tato práce je rozdělena na část teoretickou a vlastní implementaci detektoru dopravních značek.

V teoretické části jsou popsány obecné obrazové příznaky a klasifikátory, které by mohly být v tomto problému aplikovány. Ve vlastní implementaci byla experimentálně ověřena funkčnost, přesnost a rychlost detekce dopravních značek. V této části jsou popsány kroky při tvorbě aplikace na detekci dopravních značek. V závěrečné kapitole jsou uvedeny problémy, které mohou nastat zavedením této aplikace do plného provozu a rovněž jsou zde popsány klady a zápory tohoto systému.

Klíčová slova: detekce, dopravní značky, obrazový příznak, klasifikátor, HOG, SVM

Abstract

The aim of this bachelor thesis was to detect traffic signs using images in car that are equipped with cameras. With this system, we can increase road safety and can also be used to navigate self-driving vehicles. This thesis is divided into theoretical and actual implementation of traffic sign detector.

The theoretical part describes general image features and classifiers that could be applied in this problem. In its own implementation the functionality, accuracy and speed of the traffic sign detector were experimentally verified. This section describes the steps in creating a traffic sign detector. In the final chapter there are problems that can occur by introducing this application into full operation and also the pros and cons of this system.

Key Words: detection, traffic sign, image features, classifier, HOG, SVM

Obsah

Seznam použitých zkratek a symbolů	8
Seznam obrázků	9
Seznam tabulek	10
1 Úvod	11
2 Dopravní značky a datasety	12
2.1 INI Benchmark	13
2.2 BelgiumTS Dataset	14
2.3 Russian Traffic Sign Dataset	14
2.4 Swedish Traffic Sign Recognition	15
2.5 LISA Traffic Sign Dataset	15
2.6 Souhrn jednotlivých datasetů	16
3 Metody pro detekci dopravních značek	17
3.1 Lokální binární vzor	17
3.2 Histogram orientovaných gradientů	19
3.3 Detektor Viola-Jones	21
4 Klasifikátory	23
4.1 Support Vector Machines	23
4.2 Adaboost	24
5 Vlastní implementace a popis programu	25
5.1 Předzpracovávání obrazu a trénování klasifikátoru	25
5.2 Detektor	28
6 Experimenty	30
6.1 Experimenty s klasifikátorem SVM	30
6.2 Experimenty s detektorem	33
6.3 Výsledky experimentů	35
7 Experimenty na videosekvencích	36
8 Závěr	39
Literatura	40

Přílohy	41
A Příloha v IS EDISON	42

Seznam použitých zkratek a symbolů

GTSRB	– German Traffic Sign Recognition Benchmark
GTSDb	– German Traffic Sign Detection Benchmark
RTSD	– Russian Traffic Sign Dataset
STSD	– Swedish Traffic Sign Recognition
LISA	– LISA Traffic Sign Dataset
LBP	– Local Binary Pattern
DCT	– Discrete Cosine Transform
HOG	– Histogram of Oriented Gradients
LSS	– Local Self-Similarity
SVM	– Support Vector Machines
CNN	– Konvoluční neuronové sítě
TP	– True positive
FP	– False positive
FN	– False negative

Seznam obrázků

1	Svislé dopravní značky. [2]	12
2	Snímky pro testování z GTSDb. [2]	13
3	Dopravní značky, které poskytuje dataset BelgiumTS. [3]	14
4	Snímky z RTSD. [4]	14
5	Snímky z STSD. [5]	15
6	Snímky z STSD. [6]	15
7	Ilustrace základního LBP operátoru.	18
8	Lokální okolí využívaná metodou LBP. [8]	18
9	Vizualizace histogramu orientovaných gradientů pro dané dopravní značky. [21]	20
10	Posouvání bloku v obraze. [10]	21
11	Rozdělení typu obdélníků podle typu informace. [17]	22
12	Hodnota integrálního obrazu v bodě (x, y)	22
13	Nalezení optimálního oddělovače. [12]	24
14	Snímky pozadí pro zpracovávání.	26
15	Dopravní značka za různých světelných podmínek a úhlech.	26
16	Zobrazený výřez dopravní značky z obrazu pomocí souřadnic. [2]	26
17	Vizualizace příznaků HOG.	27
18	Vizualizace detekce za různé hodnoty parametru C .	31
19	Porovnání detekce za použití různé hodnoty $gamma$.	33
20	Seznam dopravních značek. [2]	36
21	Nedetekovaná dopravní značka.	37
22	Špatně označené objekty.	37
23	Dopravní značky ve větší vzdálenosti.	38
24	Dopravní značky v krátké vzdálenosti.	38

Seznam tabulek

1	Počet tříd a snímků pro jednotlivé datasety	16
2	Změna parametru C u primárního klasifikátoru a ovlivnění přesnosti detektoru. .	31
3	Změna parametru C u sekundárního klasifikátoru a ovlivnění přesnosti detektoru.	32
4	Změna parametru $gamma$ u sekundárního klasifikátoru a ovlivnění přesnosti de- tektoru.	32
5	Změna velikosti detekčního okna a ovlivnění přesnosti a rychlosti detekce. . . .	34
6	Změna koeficientu pro regulaci podobných detekcí.	34
7	Změna koeficientu pro regulaci podobných detekcí.	35
8	Výsledky detekce.	35
9	Výsledky detekce ve videích.	36

1 Úvod

Inteligentní systémy ve vozidlech mají řidiči usnadnit ovládání vozidla a zvýšit dopravní bezpečnost a komfort jízdy. Žádný inteligentní systém však z řidiče neodebírá zodpovědnost za bezpečnou jízdu vozidla.

Inteligentní dopravní systémy dělíme na IVIS a ADAS. Tyto systémy mohou řidiči poskytovat asistenci, varovat jej nebo jej plně nahrazovat v některých dílčích úkonech. Pojem IVIS (In Vehicle Information Systems) je informační systém ve vozidle a patří sem různé komunikační a navigační systémy, např: autorádio, navigace, mobilní telefon nebo palubní počítač. Systém ADAS (Advanced Driver Assistance Systems) přímo zasahuje do řízení vozidla. Má sloužit k zefektivnění řízení a ke zdokonalení práce s informacemi.

Technologie ADAS může být založena na vizuálních systémech a aktivní technologii senzorů, datové síti automobilů atd. Jednou z největších výzev systému ADAS je navigace vozidla v reálném světě.

Jedna ze součástí systému pro asistenci řízení je i systém detekce a rozpoznávání dopravních značek, který může zvýšit bezpečnost tím, že informuje řidiče o současném stavu dopravních značek na silnici a dává cenné informace o preventivních opatřeních. Při rozpoznávání dopravních značek však mnoho faktorů způsobuje problémy, jako jsou změny světelných podmínek, deformace znaků, rozostření pohybu v obrazových snímcích, překrytí dopravních značek různými překážkami atd.

Dopravní značky jsou jednoduché piktogramy určené k řízení a regulaci silničního provozu na pozemních komunikacích. Jedná se o zařízení upozorňující účastníky silničního provozu na nebezpečná místa, ukládající zákazy, příkazy nebo omezení. V reálném světě mohou řidiči ne vždy vidět dopravní značky. V noci nebo za špatného počasí jsou dopravní značky hůře viditelné a řidiči mohou být oslněni světlomety protijedoucích vozidel. Tyto situace mohou vést k těžkým dopravním nehodám.

Tato práce se zaměřuje právě na detekci dopravních značek. Důraz je kladen zejména na v posledních letech velmi populární detektory, které jsou založeny na takzvaném posuvném detekčním okně. Z obrazů získaných pomocí této techniky je poté extrahován vektor příznaku. Tento vektor je použit jako vstup pro trénovací klasifikátory (např. SVM). V teoretické části jsou popsány obecné obrazové příznaky a klasifikátory, které by mohly být v tomto problému aplikovány. Ve vlastní implementaci byla experimentálně ověřena funkčnost, přesnost a rychlost detektoru dopravních značek. V závěrečné kapitole jsou uvedeny problémy, které mohou nastat zavedením této aplikace do plného provozu a rovněž jsou zde popsány klady a zápory tohoto systému.

2 Dopravní značky a datasety

Tato kapitola se zabývá popisem svislých dopravních značek a datasety, které mohou tyto dopravní značky obsahovat. Následující informace o dopravních značeních jsou čerpány ze zákona o provozu na pozemních komunikacích [1]. Obrázek 1 zobrazující příklad svislých dopravních značek.

Rozlišují se dopravní značky svislé a vodorovné. Svislé dopravní značky jsou

- *výstražné značky, které upozorňují na místa, kde účastníku provozu na pozemních komunikacích hrozí nebezpečí a kde musí dbát zvýšené opatrnosti,*
- *značky upravující přednost, které stanoví přednost v jízdě v provozu na pozemních komunikacích,*
- *zákazové značky, které ukládají účastníku provozu na pozemních komunikacích zákazy nebo omezení,*
- *příkazové značky, které ukládají účastníku provozu na pozemních komunikacích příkazy,*
- *informativní značky, které poskytují účastníku provozu na pozemních komunikacích nutné informace, slouží k jeho orientaci nebo mu ukládají povinnosti stanovené tímto zákonem nebo zvláštním právním předpisem*
- *dodatkové tabulky, které zpřesňují, doplňují nebo omezují význam dopravní značky, pod kterou jsou umístěny.*

Prováděcí právní předpis stanoví podrobnosti dělení informativních značek.



Obrázek 1: Svislé dopravní značky. [2]

Na internetu je k dispozici celá řada datasetů pro detekci dopravních značek. Dataset je kolekce dat. Pro detekci značek se využívá pozitivní a negativní množina snímků. Tyto snímky jsou použity k učení klasifikátorů, který se použije při následné detekci. Pozitivní množina musí obsahovat objekt, který chceme detekovat v různých úhlech a za různých světelných podmínek. Negativní množina obsahuje objekt, který nechceme detekovat.

2.1 INI Benchmark

Tato instituce [2] dává k dispozici benchmark data a výsledky pro počítačové vidění a strojové učení. Obsahuje dva typy datasetu. Záznamy jsou pořízené v Německu.

German Traffic Sign Recognition Benchmark (GTSRB)

Tento dataset je určen pro rozpoznávání dopravních značek. Jedná se o kolekci dat v podobě tříd dopravních značek, kde každá třída obsahuje několik snímků. Obsahuje více jak 40 tříd dopravních značek a v konečném počtu obsahuje více jak 50 000 snímků. Snímky jsou pořízené v reálném prostředí pomocí kamery. Dataset obsahuje spolehlivé údaje pomocí poloautomatické anotace. Každý snímek je unikátní a v rámci datové sady se tedy vyskytuje pouze jednou.

German Traffic Sign Detection Benchmark (GTSDB)

Tento dataset je určen pro detekci dopravních značek. Dataset obsahuje snímky o velikosti 1360 x 800 pixelů, které jsou uložené ve formátu PPM. Dataset obsahuje rovněž soubor ve formátu CSV, který obsahuje data o snímcích, jako je název, šířka, výška a souřadnice, kde se daná značka ve snímku nachází. Velikost značky ve snímcích je od 16 x 16 pixelů až po 128 x 128 pixelů. Dopravní značky se mohou objevit v různých úhlech a za různých světelných podmínek. Obrázek 2 zobrazuje snímky pro testování z datasetu GTSDB.



Obrázek 2: Snímky pro testování z GTSDB. [2]

2.2 BelgiumTS Dataset

Další dataset dopravních značek byl autory publikován v článku [3] a obsahuje extrahované dopravní značky ze snímků pořízených pomocí kamery. Tento dataset obsahuje 62 tříd dopravních značek. K dispozici jsou čtyři video záznamy natočené pomocí osmi kamer. Dataset dále obsahuje okolo 16 000 snímků, které nemají žádnou dopravní značku. Ukázka z datasetu je na obrázku 3.



Obrázek 3: Dopravní značky, které poskytuje dataset BelgiumTS. [3]

2.3 Russian Traffic Sign Dataset

Ruský dataset (Russian Traffic Sign Dataset, RTSD) publikovali autoři v článku [4] a obsahuje snímky pro trénování a testování algoritmů pro rozpoznávání dopravních značek. Dataset obsahuje 156 tříd dopravních značek. Snímky jsou pořízené kamerou, která pořizuje 5 snímků za sekundu. Rozlišení snímků je od 1 280 x 720 pixelů až do 1 920 x 1 080 pixelů. Pořizování snímků probíhalo v různých ročních obdobích, v průběhu dne a za různého počasí. Dataset obsahuje 179 138 označených snímků, 104 358 snímků značek a 15 630 fyzických značek. Ukázka z datasetu je na obrázku 4.



Obrázek 4: Snímky z RTSD. [4]

2.4 Swedish Traffic Sign Recognition

Švédský dataset (Swedish Traffic Sign Recognition, STSD) autoři publikovali v článku [5] byl vytvořen pomocí kamery umístěné v automobilu na čelním skle, která zaznamenávala dopravní značky na území Švédska na více jak 350 km dálnic a silnic ve městě. Dataset poskytuje více jak 20 000 snímků, z toho je 3 488 snímků manuálně zpracováno. Tyto snímky obsahují dopravní značky a jejich souřadnice jsou uloženy v souboru. Soubor obsahuje typ dopravní značky, viditelnost značky a umístění dopravní značky. Ukázka snímků z datasetu je na obrázku 5.



Obrázek 5: Snímky z STSD. [5]

2.5 LISA Traffic Sign Dataset

Americký dataset (LISA Traffic Sign Dataset, LISA) byl autory publikován v článku [6] a jedná se o sadu videí a snímků obsahující dopravní značky na území USA. Zdrojová videa jsou rozdělena do skupin. Každá skupina obsahuje typ dopravní značky, kterou daná skupina reprezentuje. Skupina může obsahovat i jiné dopravní značky. Z každé skupiny je vybráno 30 snímků, které jsou zpracovávány a u všech dopravních značek je zaznamenána pozice, typ značky a další informace. Dataset obsahuje 47 tříd dopravních značek. Manuálně je označeno 7 855 dopravních značek v 6 610 snímcích o velikosti dopravních značek 6 x 6 pixelů až po 167 x 168 pixelů. Ukázka snímků z datasetu je na obrázku 6.



Obrázek 6: Snímky z STSD. [6]

2.6 Souhrn jednotlivých datasetů

Jednotlivé datasety byly porovnány z hlediska počtu trénovacích snímků, celkového počtu snímků a počtu tříd dopravních značek. Tabulka 1 zobrazuje souhrn informací ohledně datasetů uvedených výše. Nejlépe z nich vychází dataset GTSRB 2.1, který disponuje mnoha snímky pro trénování i pro testování. Oproti tomu GTSDb 2.1 obsahuje o mnoho méně snímků pro testování i trénování, ale je to dáno tím, že dataset je primárně určen pro detekci dopravních značek.

Další dataset, který obsahuje jak snímky pro testování, tak i pro trénování, je BelgiumTS 2.2. Tento dataset poskytuje i snímky, kde není vyobrazena žádná dopravní značka, a videa pro testování.

U datasetu RTSD 2.3 není uvedeno, kolik snímků je určeno k testování nebo trénování. Obsahuje obrovské množství snímků s dopravními značkami a ty jsou rozděleny do 156 tříd, což je velmi vysoký počet tříd oproti jiným datasetům.

Dataset LISA 2.5 a STSD 2.4 obsahují nejen informace o souřadnicích dopravní značky, ale i informace o viditelnosti a umístění dopravní značky na okraji vozovky. Tyto dva datasety nemají uvedené, zda obsahují trénovací nebo testovací snímky.

Tabulka 1: Počet tříd a snímků pro jednotlivé datasety

Název	Počet tříd	Počet trénovacích snímků	Celkový počet snímků
GTSRB	43	39 209	50 000
GTSDb	43	1 213	2 113
BelgiumTS	62	4 591	25 630
RTSD	156	není uvedeno	179 138
STSD	19	není uvedeno	20 000
LISA	47	není uvedeno	6 610

3 Metody pro detekci dopravních značek

Účelem detekce dopravních značek je nalezení místa a rozměrů dopravních značek v přírodních scénériích. Pro tuto detekci jsou důležité dobře definované barvy a tvary těchto značek. V této práci je věnován větší prostor metodě HOG, která je využita ve vlastní implementaci. Metody detekce lze rozdělit do dvou kategorií [7]: detekci na základě barvy a detekci na základě tvaru. Metody na základě detekce barvy jsou obvykle rychlejší a nepodléhají transformaci, rotaci a změně velikosti obrazu.

Metody detekce na základě barvy

Barva může být jednoduše ovlivněna světlem. Největší problém metody detekce na základě barvy je stabilita barevnosti při změně světelných podmínek. Tyto metody jsou založeny na obecném postupu přeměny obrazu do barevného prostoru a poté do prahových hodnot. Další možností je převod prahové hodnoty rovnou do RGB schématu, ale tato metoda je velmi citlivá na změny osvětlení. Lze využít i YUV a HSI schéma.

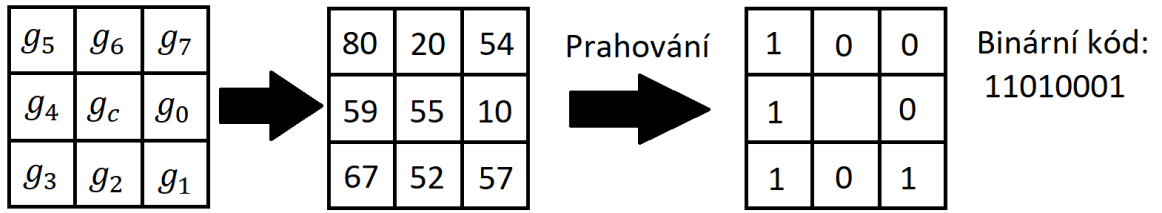
Metody detekce na základě tvaru

Metody na základě detekce tvaru mohou používat např. Haarovy příznaky inspirované Viola-Jones detektorem [9], příznaky reprezentované histogramy LBP [8] nebo histogramy orientovaných gradientů v obrazech HOG [14]. Další metodou je konvoluční neuronová síť (CNN). Jedná se o nejúspěšnější metodu strojového učení současnosti. Neuronové sítě se snaží o jedno - co nejvíce zkvalitnit schopnost čtení a pochopení obrazu počítačem.

3.1 Lokální binární vzor

Lokální binární vzor (Local Binary Pattern, LBP) představil Timo Ojala [8] v roce 2002. Metoda byla vytvořena za účelem sestrojení jednoduchého a výpočetně rychlého nástroje pro klasifikaci textur v obraze. Metoda není závislá na změně intenzity jasu, kontrastu a rotaci obrazu. Následující text byl čerpán z publikací [8] a [10].

Standardní verze metody LBP jednoho pixelu se získá pomocí prahování všech pixelů s hodnotou středového pixelu na okolí o velikosti 3×3 . Nechť je g_c střed pixelu šedého odstínu a $g_i (i = 0, 1, \dots, 7)$ je úroveň šedi každého okolního pixelu. Obrázek 7 vizualizuje základní LBP operátor. Pokud je g_i menší než g_c , tak binární výsledek pixelu je nastaven na 0, v opačném případě je nastaven na 1. Všechny výsledky jsou kombinovány k dosažení 8 bitové hodnoty. Binární hodnota desítkové soustavy je LBP příznak.

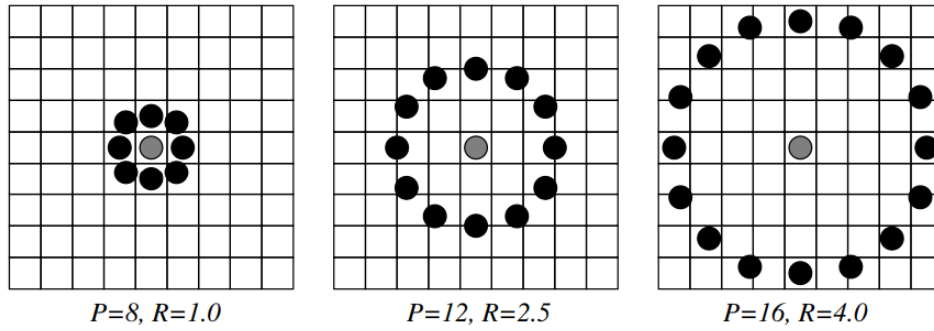


Obrázek 7: Ilustrace základního LBP operátoru.

Bilineární interpolační metoda se využívá pro vzorkování bodu, které nepatří do středového pixelu. Nechť $LBP_{P,R}$ označuje LBP příznak kruhové reprezentace, kde R je poloměr a P je počet sousedících bodů na kruhu.

$$LBP_{P,R} = \sum_{p=0}^{p-1} s(g_i - g_c)2^i, S(x) = \begin{cases} 1 & \text{pokud } x \geq 0 \\ 0 & \text{jinak} \end{cases} \quad (1)$$

Obrázek 8 zobrazuje lokální okolí pro různé hodnoty parametrů P a R .



Obrázek 8: Lokální okolí využívaná metodou LBP. [8]

Koncept uniformních binárních vzorů byl založen na redukování počtu binárních vzorů. Každý LBP vzor se nazývá uniformním, pokud se binární vzor skládá nejvýše ze dvou bitových přechodů z 0 na 1 nebo naopak. Například binární vzor 11111111 je uniformní, protože neobsahuje žádné přechody. Naopak binární vzor 10101011 obsahuje 6 přechodů a nejedná se o uniformní binární vzor. Uniformní binární vzory redukují počet LBP vzorů od 256 do 58 a je velmi užitečný při detekci tváře [23]. Zvýšení robustnosti LBP metody se docílí tím, že výsledný deskriptor není složen ze všech nalezených binárních vzorů. Množina vzorů je před následným zpracováváním zredukována na vzory pouze uniformní.

Metoda LBP se využívá i při detekci dopravních značek. Nový přístup pro detekci dopravních značek založený na využití vlastností LBP a DCT (Discrete Cosine Transform) metod navrhuji autoři v článku [20]. Tyto metody jsou vhodné pro extrakci obrazů, vzájemně se doplňují a pomocí metody SVM (Support Vector Machines) dochází ke zlepšení detekce. V dalším článku [19] se autoři zabývají detekcí a rozpoznáváním dopravních značek pomocí deskriptoru HOG-

LBP. Jako klasifikátor byl použit Random forest. Navrhované řešení má dobré výsledky při detekci dopravních značek za zhoršených světelných podmínek.

3.2 Histogram orientovaných gradientů

Histogram orientovaných gradientů (Histograms of Oriented Gradients, HOG) je deskriptor používaný v počítačovém vidění a zpracování obrazu pro účely detekce objektů. Metoda HOG byla vyvinuta v roce 2005 za účelem detekce lidí v obraze. Pracuje na principu popisu objektu za pomoci distribuce gradientů. Jednotlivé gradienty jsou dány velikostí a směrem. Vstupem metody HOG může být obraz barevný nebo šedo tónový. Tyto obrazy jsou pak za účelem zvýšení robustnosti normalizovány pomocí gamma korekce.

V článku [7] autoři publikovali využití metody HOG k detekci dopravních značek. Detekce dopravních značek se skládá ze tří částí. Prvním krokem je prahování pomocí HSV. Dalším krokem je detekce kruhových, trojúhelníkových a obdélníkových tvarů na segmentovaných snímcích pomocí klasifikátoru Random forest. Poslední krok spočívá v identifikaci detekované dopravní značky. Jsou porovnávány čtyři deskriptory funkcí HOG, Gabor, LBP a LSS (Local Self-Similarity). Nejlepších výsledků bylo dosaženo kombinací HOG s LSS a Random forest klasifikátorem. Následující text byl čerpán z článků [10] a [14] a zabývá se výpočtem histogramu orientovaných gradientů a konstrukcí HOG deskriptoru.

Výpočet gradientu

Pro výpočet gradientu bylo otestováno mnoho variant stanovení gradientu v obraze. Prvním krokem je filtrování obrazu pomocí Gaussového jádra o různém parametru σ . Následně byly aplikovány masky pro výpočet diskretního gradientu. Výsledky testu byly vyhodnoceny a pro metodu HOG byly stanoveny nejlepší kombinace parametru σ a filtrační masky. Metoda HOG získává gradienty pomocí konvoluce Gaussovsky filtrovaného obrazu I při $\sigma=0$ s maskou $[-1, 0, 1]$ respektive $[-1, 0, 1]^T$, kde $*$ značí konvoluci.

$$I_x = I * [-1, 0, 1] \quad (2)$$

$$I_y = I * [-1, 0, 1]^T \quad (3)$$

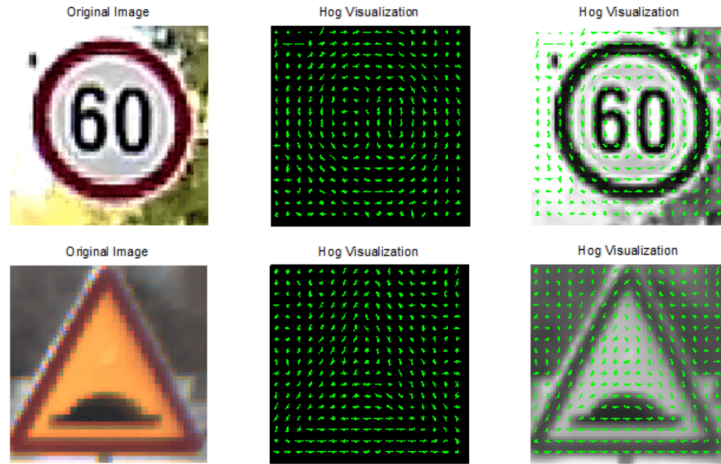
Histogram orientací gradientů

Po získání obrazu I_x a I_y je pro každý pixel vypočtena velikost $m(x, y)$ a směr gradientu $\Theta(x, y)$.

$$m(x, y) = \sqrt{I_x^2 + I_y^2} \quad (4)$$

$$\Theta(x, y) = \tan^{-1}\left(\frac{I_y}{I_x}\right) \quad (5)$$

Počet binů histogramu orientací je daný intervalem i . Interval i je rozsah, kde se zjišťuje směr gradientu $\Theta(x, y)$. Nejvíce používané rozsahy jsou $i = \langle 0, \pi \rangle$, $i = \langle 0, 2\pi \rangle$. Přínos vypočtených gradientů do jednotlivých binů histogramu je určen jejich velikostmi $m(x, y)$ a navíc je binárně rozdělen i mezi biny sousední. Tím se zvýší robustnost výsledného deskriptoru. Obrázek 9 zobrazuje histogramy orientovaných gradientů vytvořené pro dopravní značku.



Obrázek 9: Vizualizace histogramu orientovaných gradientů pro dané dopravní značky. [21]

Normalizace histogramu

Z důvodu dosažení invariance deskriptoru vůči jasovým transformacím obrazu (změna jasu a kontrastu) jsou získané histogramy orientací normalizovány. Nejvíce využívaná normalizační schémata jsou $L^1 - norm$, $L^1 - sqrt$ a $L^2 - norm$.

$$L^1 - norm = \frac{v}{(\|v\|_1 + e)} \quad (6)$$

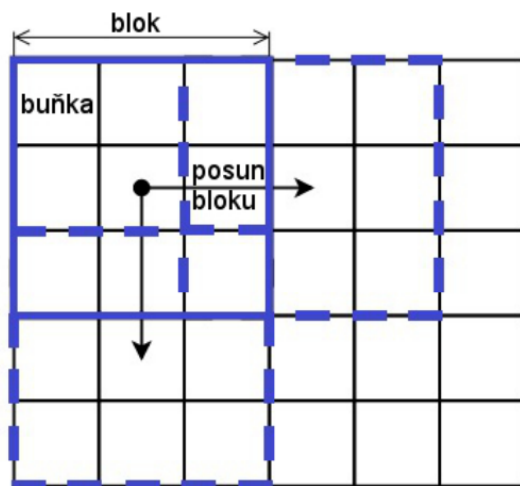
$$L^1 - sqrt = \sqrt{\frac{v}{(\|v\|_1 + e)}} \quad (7)$$

$$L^2 - norm = \frac{v}{(\|v\|_2^2 + e^2)} \quad (8)$$

Normalizovaný vektor je v , $\|v\|_k$ představuje jeho k -tou normu a e je malá konstanta.

Konstrukce deskriptoru

Deskriptor HOG je získán rovnoměrným rozdělením vstupního obrazu do čtvercových bloků. Poté jsou bloky rozčleněny na buňky, pro které jsou vypočteny normované histogramy orientací gradientů. Výsledné histogramy buněk tvoří HOG popis bloku, po jehož získání se blok posune o zvolený počet pixelů a celý proces se opakuje, dokud nedojde na konec obrazu. Finální HOG deskriptor je zřetěžením všech popisů jednotlivých bloků uvnitř vstupního obrazu. Obrázek 10 vizualizuje posun bloku v obraze.



Obrázek 10: Posouvání bloku v obraze. V konkrétním případě obsahuje blok 9 buněk. [10]

3.3 Detektor Viola-Jones

Detektor Viola-Jones je metoda, která byla představena v roce 2001 pro detekci obličejů. V dalších letech se ukázalo, že může být použita i pro detekci jiných objektů (chodců, automobilů, dopravních značek). Výhodou tohoto detektoru je rychlost detekce a díky tomu lze provádět detekci v reálném čase. O existenci objektu v obraze rozhoduje obecný detekční algoritmus založený na výpočtu informativních příznaků. Detektor využívá jako příznaky odezvu na Haarovy filtry. K trénování detektoru slouží klasifikační algoritmus AdaBoost. Algoritmus se skládá z jednoduchých klasifikátorů. Pro vytvoření klasifikátoru je potřeba obrovské množství pozitivních a negativních vzorků. Detekce je založena na výběru výřezu ze vstupního obrazu, ze kterého jsou vypočteny příznaky. Poté je výřez klasifikován jako hledaný objekt nebo pozadí. Velikost výřezu odpovídá velikosti objektu ve vstupním obrazu. Aby bylo možné detekovat objekty o obecném rozměru, musí se změnit rozměr výřezu detektoru. Velikost vybraného výřezu odpovídá velikosti objektu ve vstupním obraze. Následující text byl čerpán z článků [9], [15], [16] a popisuje princip fungování Haarových příznaků.

3.3.1 Haarovy příznaky

Haarových příznaků se využívá hlavně při tvorbě klasifikátoru pro detekci v obrazech v metodě Viola-Jones [9]. Tyto příznaky jsou tvořeny obdélníky, které jsou bílé nebo černé. Pro výpočet hodnoty příznaku je nutné spočítat rozdíl mezi bílými a černými obdélníky. Než se tyto obdélníky od sebe odečtou, je nutné spočítat sumu intenzity všech pixelů jednotlivých obdélníků. Tyto obdélníky se dělí podle typu informace na hranové, čárové a příznaky středové. Obrázek 11 zobrazuje jednotlivé obdélníky na dopravní značce.



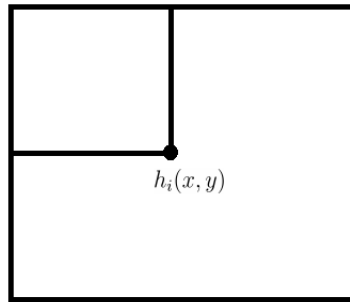
Obrázek 11: Rozdělení typu obdélníků podle typu informace. [17]

3.3.2 Integrální obraz

Pomocí obdélníkových příznaků dojde k výrazně vyšší rychlosti výpočtu odezvy příznaku. Účelem převedení vstupního obrazu na obraz integrální je dosažení vyšší rychlosti výpočtů odezvy příznaků. Zvýšení rychlosti výpočtu dosáhneme tak, že se nebude pokaždé počítat součet hodnot pixelů v definovaných obdélnících u vstupního obrazu. Hodnota na souřadnicích $h_i(x, y)$ v integrálním obrazu se vypočítá podle vzorce:

$$h_i(x, y) = \sum_{x' \leq x, y' \leq y} h(x', y'), \quad (9)$$

kde $h_i(x, y)$ jsou hodnoty integrálního obrazu a $h(x, y)$ jsou hodnoty pixelů vstupního obrazu.



Obrázek 12: Hodnota integrálního obrazu v bodě (x, y)

4 Klasifikátory

V současné době roste počet automobilů v dopravním provozu. Dopravní značky jsou velice důležité pro zajištění bezpečnosti a plynulosti provozu. Bohužel se stává hodně dopravních nehod z důvodů přehlédnutí dopravní značky řidičem. Proto je stále více automobilů vybaveno kamerovým systémem, který detekuje dopravní značky za pomoci obrazů pořízených kamerou. K této detekci a následnému rozpoznání mohou být použity různé techniky z oblasti zpracování obrazů. Jedna z možných technik je použití kombinace obrazových příznaků a trénovacího klasifikátoru např: neuronová síť nebo SVM. Dalšími klasifikátory mohou být Random forest, kaskádový klasifikátor a Adaboost.

Algoritmus rozpoznávání snímku (obrazový klasifikátor) zachycuje obraz jako vstup a výstup, jinými slovy, co se na tom obrazu nachází. Pokud chceme najít určitý objekt v obrazech, musíme vycvičit algoritmus rozpoznávat obrázky s objektem, který chceme najít a obrazy pozadí, které tyto objekty neobsahují. Tento algoritmu může nacházet pouze objekty, které se naučil.

4.1 Support Vector Machines

Podpůrné vektory (Support Vector Machines, SVM) patří do skupiny algoritmů pro strojové učení. Tvoří určitou kategorii tzv. jádrových algoritmů (kernel machines). Jádrové funkce využívají výhody poskytnuté efektivními algoritmy pro nalezení lineární hranice a zároveň jsou schopny reprezentovat vysoce složité nelineární funkce. Následující text byl čerpán z článků [11] a [12].

Základní princip SVM je převod daného původního prostoru do jiného vícedimensionálního, kde lze třídy oddělit lineárně. Ve dvourozměrném vstupním prostoru jsou definované atributy $x = (x_1, x_2)$. Pro pozitivní trénovací příklady, dávají hodnotu +1 a pro negativní -1.

Jádrové funkce umožňují najít optimální lineární oddělovač. Optimální lineární oddělovač poskytuje co nejširší pásmo mezi negativními příklady na jedné straně a pozitivními na straně druhé a zároveň podporuje robustnost klasifikace. Pomocí metody kvadratického programování lze najít optimální lineární oddělovač. Jistá obdoba je u lineárního programování s hledáním maxima. Pro nalezení optimálního oddělovače je nutné hledání hodnot parametrů α_i , které maximalizují výraz

$$\sum_i a_i - \frac{1}{2} \sum_{i,j} \alpha_i, \alpha_j, y_i, y_j (x_i \cdot x_j) \quad (10)$$

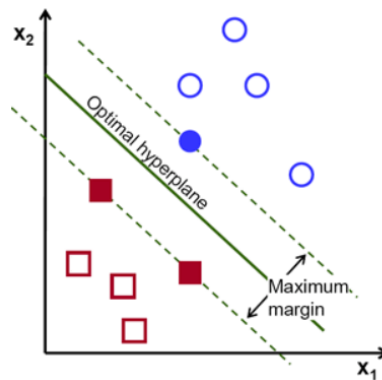
přičemž musí platit omezení

$$\alpha_i \geq 0, \sum_i \alpha_i, y_i = 0 \quad (11)$$

Výraz má jediné globální maximum, které lze efektivně najít. Datové body vstupují do výrazu ve formě skalárního součinu jednotlivých dvojic, což platí i pro rovnici lineárního oddělovače. Pokud jsou spočteny optimální parametry α_i je oddělovač dán rovnicí

$$h(x) = \text{sign}\left(\sum_i \alpha_i y_i (x \cdot x_i)\right) \quad (12)$$

Při hledání α_i platí, že lineární oddělovač má nulové váhy α_i pro každý datový bod kromě bodů, které jsou nejbližší vlastnímu oddělovači. Body, které se nachází nejbližší, se nazývají podpůrné vektory (support vectors), které podporují oddělovací nadrovinu. Ostatní body-vektory nejsou pro oddělovače potřeba. Metoda SVM je schopna najít ty trénovací příklady, které jsou pro nalezení oddělovače tříd velmi důležité. Obrázek 13 znázorňuje nalezení optimálního oddělovače pomocí podpůrných vektorů.



Obrázek 13: Nalezení optimálního oddělovače. [12]

4.2 Adaboost

Algoritmus Adaboost (Adaptive boosting) byl vyvinut v roce 1995 a jedná se o klasifikační algoritmus. Jedná se o metodu, která umožňuje vhodné spojení několika klasifikátorů s relativně špatnou úspěšností, tzv. slabých klasifikátorů do jednoho tzv. silného klasifikátoru. Výhodou tohoto klasifikátoru je jeho rychlost. Algoritmus Adaboost lze rovněž použít při velkém množství příznaků.

Adaboost vybírá jednotlivé informativní příznaky podle toho, jakou mají chybovost při klasifikaci obrazů. Klasifikátor je možné natrénovat na detekci odlišných vzorů v obraze. V jednotlivých cyklech se provádí nastavení důležitosti správné klasifikace jednotlivých obrazů: váha obrazů, výpočet příznaků z trénovací množiny, nastavení optimálního prahu Θ minimalizující chybovost slabého klasifikátoru používající daný příznak, výběr klasifikátoru s nejmenší chybovostí a jeho zařazení mezi klasifikátory použité při detekci.

Slabý klasifikátor může být například neuronová síť, rozhodovací strom nebo i jeden Haarův příznak. Podmínka pro zařazení slabého klasifikátoru je ta, že musí mít úspěšnost vyšší než 50%.

5 Vlastní implementace a popis programu

V této kapitole je popsána implementace samotného programu pro trénování klasifikátoru a následná detekce dopravních značek v obrazech nebo videu. Taktéž zde bude popsáno předzpracovávání pozitivní a negativní množiny pro klasifikátor. Popis a implementaci programu jsem rozdělil do dvou částí. První část obsahuje popis předzpracovávání obrazu a trénování klasifikátoru. Druhá část bude zaměřena na detekci dopravních značek. Program může přeskočit první část, pokud je klasifikátor již vytvořen a může ho načíst z disku. Jako programovací jazyk jsem zvolil C++, který je plně podporován knihovnou OpenCV. Jazyk C++ plně podporuje objektové programování pro zlepšení efektivity při návrhu programu.

Knihovna OpenCV

OpenCV (Open Source Computer Vision Library) je open source knihovna, které se využívá při práci se strojovým učením a počítačovou vizí. Byla vytvořena tak, aby poskytovala společnou infrastrukturu pro aplikace zaměřující se na počítačové vidění a urychlila používání strojového učení v komerčních produktech. Knihovna OpenCV disponuje kódy, které lze jednoduše upravit a využít.

5.1 Předzpracovávání obrazu a trénování klasifikátoru

Nutné vstupy pro trénování klasifikátoru jsou adresáře obsahující pozitivní a negativní obrázky. Další vstup je šířka a výška obrazu pomocí pixelů. Všechny pozitivní a negativní obrazy musí mít stejnou šířku a výšku. Šířku a výšku obrazu jsem zvolil 32 x 32 pixelů, aby bylo možné detekovat i značky, které jsou ve větší vzdálenosti od kamery.

5.1.1 Předzpracovávání negativních obrazů

Pro trénování klasifikátoru je potřeba získat negativní množinu obrazů. Jako negativní množinu obrazů jsem využil dataset BelgiumTS 2.2, který má k dispozici obrazy pozadí bez žádné dopravní značky. Tyto obrazy mají velikost 1 628 x 1 236 pixelů a jsou uloženy ve formátu jp2. Program postupně načítá jednotlivé obrazy a kontroluje, zda nejsou poškozeny. Pokud je obraz poškozen, program jej přeskočí. Pomocí parametru se dá nastavit počet obrazů, které se mají vyříznout z originálního obrazu na velikost zadanou při vstupu, což je 32 x 32 pixelů. Výběr je zcela náhodný. Tento náhodný výběr se provede jen jednou a pak se s tímto výběrem obrázků dále pracuje. Výsledné obrazy jsou uloženy v odděleném adresáři. Na obrázku 14 jsou zobrazeny snímky, z kterých jsou následně vyřezány obrazy o velikosti 32 x 32 pixelů.



Obrázek 14: Snímky pozadí pro zpracování.

5.1.2 Předzpracování pozitivních obrazů

I pozitivní množinu obrazů je nutné předzpracovat pro trénování klasifikátoru. Pro pozitivní obrazy jsem využil dataset GTSDb 2.1 a GTSRB 2.1. Oba datasety obsahují 43 adresářů a v každém adresáři jsou obrazy stejných dopravních značek, ale v různých úhlech a světelných podmínkách. Obrázek 15 zobrazuje dopravní značku v různých úhlech a světelných podmínkách.



Obrázek 15: Dopravní značka za různých světelných podmínek a úhlech.

Pro každý obraz jsou v souboru CSV uloženy informace pro zpracování. Tyto informace jsou: název obrazu, šířka, výška, souřadnice dopravní značky a id značky. Na obrázku 16 je zobrazen příklad výřezu dopravní značky pomocí souřadnic. Souřadnice jsou uvedeny v pixelech a udávají pozici značky v obrazu.



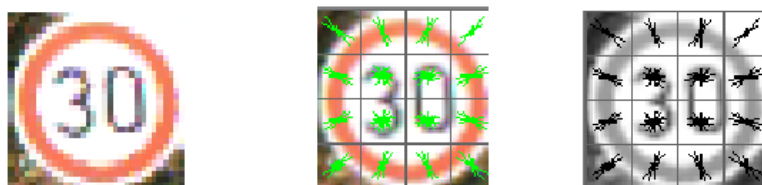
Obrázek 16: Zobrazený výřez dopravní značky z obrazu pomocí souřadnic. [2]

Program načte všechny obrazy a zkontroluje, zda nejsou poškozené. Pokud dataset neobsahuje souřadnice dopravní značky, pak se následující krok vynechá. Jestliže adresář obsahuje soubor CSV, hodnoty se použijí pro vyříznutí dopravní značky z obrazu. Jelikož obrazy do-

pravních značek mají různou velikost, musí program pro všechny obrazy nastavit fixní velikost zadanou při vstupu, což je 32 x 32 pixelů.

5.1.3 Převod na příznaky typu HOG

Po předzpracování obrazu je nutné převést obrazy do odstínu šedi a poté jsou tyto obrazy extrahovány na příznaky typu HOG. Program převede obraz z barevného modelu RGB do odstínu šedi pomocí funkce *cvtColor*, která je součástí OpenCV. Funkce má tyto parametry: vstupní obraz, výstupní obraz a typ transformace. Pro převod obrazů na odstín šedi je použit typ transformace *COLOR_BGR2GRAY*. Následně pomocí funkce *compute*, která je součástí třídy *HOGDescriptor*, jsou všechny obrazy pozitivní a negativní množiny extrahovány na příznaky typu HOG. Výstupem funkce pro každý obraz je pole hodnot, které reprezentuje příznaky HOG. Na obrázku 17 je zobrazen originální obrázek a následně obrázek vizualizace příznaků HOG v RGB modelu a odstínu šedi.



Obrázek 17: Vizualizace příznaků HOG.

5.1.4 Trénování klasifikátoru

Jako klasifikátor dopravních značek jsem použil klasifikátor SVM 4.1.

Primární klasifikátor slouží k hledání dopravních značek v celém vstupním obraze. Sekundární klasifikátor slouží ke kontrole obrazů, které primární detektor detekoval. Rozdíl mezi primárním a sekundárním klasifikátorem je v použitém jádru. Klasifikátor SVM disponuje mnoha jádry. Nejvíce používaná jádra v knihovně OpenCV jsou: lineární, POLY, RBF, SIGMOID, CHI2 a INTER. Primární klasifikátor používá jádro lineární. Lineární jádro nepoužívá žádné mapování. Využívá lineární rozlišování v původním prostoru a jedná se o jedno z nejrychlejších jader. Toto je velmi důležité, protože je primární klasifikátor využit k detekci celého vstupního obrazu. Oproti tomu sekundární klasifikátor využívá jádro RBF, které je pomalejší a nemá lineární oddělovač. Sekundární klasifikátor slouží ke kontrole obrazů z primárního detektoru, kterých je minimum.

Klasifikátory je třeba naučit pomocí řádkové matice. Jednotlivé řádkové matice jsou naskládány na sebe (každý řádek reprezentuje jeden obraz ze vstupu). Tímto způsobem je vytvořena

matice pro trénování. Metoda *train* slouží k trénování klasifikátoru. Vstupní parametry pro trénování jsou: matice s daty obrazů, rozložení dat v matici a identifikace každého obrazu v matici. Trénování klasifikátoru lze ovlivnit několika parametry. Parametry, které slouží k optimalizaci klasifikátoru SVM, jsou *gamma* a *C*.

Parametr *C*

V případě, že data nejsou dobře oddělitelná a rozhodovací rovina je příliš malá, tak dochází k chybám při trénování. V takových případech se snaží SVM najít nejlepší možnou rozhodovací rovinu tím, že vyřeší optimalizační problém. Ten spočívá ve snaze zvětšit vzdálenost rozhodovací roviny od obou tříd, ale přitom si dává pozor nato, aby trénovací příklady byly klasifikovány správně. Tuto vzdálenost lze ovlivnit parametrem *C*. Pokud je hodnota parametru *C* malá, rozhodovací rovina s velkou marží je vybrána na úkor většího počtu chyb v klasifikaci. Jestliže je hodnota parametru *C* větší, je zvolena menší marže rozhodovací roviny, která se snaží klasifikovat mnoho dalších trénovacích příkladů správně.

Parametr *gamma*

Parametr *gamma* definuje, jak velký dopad má jednotlivý trénovací příklad na trénování dat. S nižší hodnotou je dopad vyšší a naopak pokud je hodnota vyšší, pak je dopad nižší na trénovací data. Jestliže je hodnota *gamma* příliš vysoká, tak rozhodovací rovina bude ovlivněna pouze vektory, které se nachází blízko rozhodovací roviny. To pak může vést k vynechání vektorů, které se vyskytují ve větší vzdálenosti od roviny. V opačném případě, jestliže je hodnota malá, tak rozhodovací rovina bude ovlivněna i vektory, které jsou ve větší vzdálenosti od roviny.

Uložení klasifikátoru

Po ukončení trénování obou klasifikátorů jsou data uložena do adresáře ve formátu YAML. Uložení dat do souboru je velmi výhodné, jelikož při dalším spuštění se nemusí klasifikátory znovu trénovat, ale mohou se rovnou načíst. Trénování klasifikátoru je mnohonásobně pomalejší než načtení souboru. Jestliže chceme změnit parametry SVM nebo změnit počet pozitivních nebo negativních obrazů, je nutné klasifikátory přetrénovat.

5.2 Detektor

Vstupní parametry pro detektor dopravních značek jsou primární a sekundární klasifikátor, šířka a výška obrazu pro detekci v pixelech a adresář s obrazy pro hledání dopravních značek nebo adresář s videem.

Pokud se jedná o video, program načítá postupně každý snímek z videa. Jelikož video může mít několik desítek snímků za sekundu, je nutné pár snímků vynechat, protože samotná detekce trvá určitý čas. Čas strávený detekcí je závislý na výkonu procesoru. Proto jsem implementoval časovač, který umožní ve videu vynechat určitý počet snímků podle výkonu procesoru. Jestliže video má 30 snímků za sekundu a procesor zvládne zpracovat pouze 10 snímků za sekundu, tak program vynechá každý třetí snímek. Procesor by neměl být moc pomalý, aby nedošlo k přehlédnutí dopravních značek a nedošlo k vynechání nadměrného množství snímků ve videu.

Pro detekci dopravních značek v obrazech není nutné mít časovač tak jak u videa. Obrazy se načítají postupně. Následně je obraz předán funkci pro detekci dopravních značek.

Pro detekci dopravních značek je nutné detektoru nastavit klasifikátor. Pomocí funkce *setSVMDetector* nastavím detektoru primární klasifikátor SVM. Hlavní funkce, která detekuje dopravní značky v obraze, je *detectMultiScale*, která je součástí třídy *HOGDescriptor*. Tato funkce má několik vstupů:

- vstupní obraz, ve kterém se mají detekovat dopravní značky,
- hodnota, o kolik se má detekční okno zvětšit oproti původní velikosti v procentech,
- koeficient pro regulaci podobných detekcí v okolí (pokud je objekt detekován více než jednou, jsou nalezené detekce spojeny v jednu).

Výstupem funkce jsou souřadnice nalezené detekce a velikost odhadu nalezení. Čím větší je hodnota nalezení, tím větší je pravděpodobnost, že byla dopravní značka nalezena. Dalším krokem je kontrola nalezených detekcí pomocí sekundárního klasifikátoru. Kontrola probíhá pomocí funkce *predict*. Tato funkce je součástí třídy SVM. Pro kontrolu nalezených detekcí je nutné obraz předpřipravit. Nalezené souřadnice detekované značky se vyříznou z originálního obrazu. Šířka a výška se nastaví na velikost zadanou při vstupu, tedy 32 x 32 pixelů. Následně proběhne extrahování výřezu na příznaky typu HOG. Funkce poté rozhodne, zda se jedná o dopravní značku hodnotou 1. Pokud se nejedná o dopravní značku, funkce vrátí hodnotu -1.

6 Experimenty

Tato kapitola se zabývá experimenty detektoru, klasifikátoru a vylepšení přesnosti výsledků detektoru pro detekci dopravních značek. Při provádění experimentů byl použit procesor i7-9700K. Pro experimenty jsem využil obrazy pro testování. Dataset GTSDDB 2.1 obsahuje 900 snímků s dopravními značkami v různých úhlech a za různých světelných podmínek v reálném prostředí za účelem testování. Snímky jsou pořízené z kamery umístěné v automobilu. Tento dataset jsem si upravil tak, aby velikost značky byla větší než 28 x 28 pixelů.

Každý obraz může obsahovat několik dopravních značek nebo nemusí obsahovat žádnou dopravní značku. V 693 obrazech se nachází 789 dopravních značek. Každá dopravní značka v obrazu je zaznamenána v souboru, který obsahuje tyto informace: název obrazu a souřadnice, kde se dopravní značka nachází v obrazu. Pomocí těchto souřadnic se vypočítá střed dopravní značky a porovná se s výsledky detektoru. Výstupem detektoru jsou souřadnice, u kterých se také vypočítá střed. Souřadnice středu se porovnají. Pokud se střed liší o více než 5 pixelů, nejedná se o dopravní značku. Jestliže jsou si podobné do pěti pixelů, jedná se o dopravní značku.

Pro výpočet přesnosti programu jsem využil metodu pro měření přesnosti F1 score. Tato metoda se využívá ve statistické analýze k binární klasifikaci měřítka přesnosti testu. Pro výpočet F1 score je následující výraz:

$$F_1 = \frac{2TP}{2TP + FP + FN} \quad (13)$$

kde parametry výrazu jsou:

- TP – počet správně nalezených dopravních značek
- FP – počet označení, kdy se nejedná o dopravní značku
- FN – počet nedetekovaných dopravních značek

Výsledkem je hodnota v rozmezí 0-1. Hodnota 0 znamená, že úspěšnost je 0 %. Oproti tomu hodnota 1 znamená že úspěšnost je 100 %.

6.1 Experimenty s klasifikátorem SVM

Tato podkapitola se zabývá experimenty s klasifikátory SVM - změnou parametrů klasifikátoru pro trénování pozitivní a negativní množiny a jejího vlivu na přesnost detektoru. Počet pozitivních obrazů pro trénování je 1 213 z datasetu GTSDDB 2.1 a počet negativních obrazů je 2 000. Detektor HOG je nastaven tak, že detekční okno detektoru se bude zvětšovat o 5% a koeficient pro regulaci podobných detekcí v okolí je nastaven na 1.

Parametry, které ovlivňují trénování klasifikátoru SVM, jsou *gamma* a *C*. Konkrétně u primárního klasifikátoru, který je lineární, se jedná pouze o parametr *C*. Parametr ovlivňuje rozhodovací rovinu tím, že vybírá určitý počet příkladů z každé třídy v závislosti na velikosti

parametru C . V tabulce 2 jsou výsledky přesnosti detektoru v závislosti na změně parametru C u primárního klasifikátoru. Parametry pro trénování sekundárního klasifikátoru jsou $C = 0.1$ a $gamma = 0.9$.

Tabulka 2: Změna parametru C u primárního klasifikátoru a ovlivnění přesnosti detektoru.

Hodnota C	Přesnost (%)	TP	FP	FN
0.05	74.68	562	154	227
0.1	75.02	584	184	205
0.2	77.87	600	152	189
0.3	78.60	593	127	196
0.5	75.46	574	159	215
1	76.44	592	168	197

Z výsledků vyplývá, že u primárního klasifikátoru parametr C při hodnotě 0.3 vykazuje nejvyšší zaznamenanou úspěšnost. Celkem bylo nalezeno 593 dopravních značek. Nenalezeno bylo 196 značek. Celkem 127 objektů bylo nesprávně označeno za dopravní značku. Obrázek 18 zobrazuje rozdíl mezi hodnotou $C = 0.05$ na levé straně a hodnotou $C = 0.3$ na pravé straně. Pokud je hodnota C vyšší, je detekováno více objektů, které mohou být označeny za dopravní značku.



Obrázek 18: Vizualizace detekce za různé hodnoty parametru C . Černo-zelené obdélníky vizualizují všechny nálezy detektoru.

U sekundárního klasifikátoru je jádro RBF. Pro trénování tohoto jádra je zapotřebí i parametr $gamma$. Parametr $gamma$ ovlivňuje, jaký dopad mají jednotlivé trénovací příklady na trénovací data. Následující tabulka 3 zobrazuje výsledky přesnosti detektoru v závislosti na změně parametru C u sekundárního klasifikátoru. Parametry pro trénování primárního klasifikátoru jsou $C = 0.3$ a u sekundárního klasifikátoru je nastaven parametr $gamma$ na 0.9.

Tabulka 3: Změna parametru C u sekundárního klasifikátoru a ovlivnění přesnosti detektoru.

Hodnota C	Přesnost (%)	TP	FP	FN
0.05	71.44	484	82	305
0.08	77.16	566	112	223
0.1	78.60	593	127	196
0.2	78.86	647	171	164
0.3	75.94	655	281	134
0.5	74.51	668	336	121
2.0	70.30	683	463	114

U sekundárního klasifikátoru je dosažena nejvyšší úspěšnost kolem hodnoty $C = 0.1$. Při stoupající hodnotě C vzrůstá počet špatně označených dopravních značek, ale klesá počet nedetekovaných dopravních značek. Tabulka 4 zobrazuje výsledky přesnosti detektoru v závislosti na změně parametru $gamma$ u sekundárního klasifikátoru. Parametry pro trénování primárního klasifikátoru jsou $C = 0.3$ a u sekundárního klasifikátoru $C = 0.15$.

Tabulka 4: Změna parametru $gamma$ u sekundárního klasifikátoru a ovlivnění přesnosti detektoru.

Hodnota $gamma$	Přesnost (%)	TP	FP	FN
0.1	51.94	703	1215	86
0.5	65.47	695	639	94
0.8	77.24	650	244	139
0.9	78.86	625	171	164
1.0	78.36	592	130	197
1.2	75.48	528	82	261
1.5	69.15	437	38	352

Výsledky změny parametru $gamma$ ukazují, že nejvyšší úspěšnost byla zaznamenána v hodnotě 0.9. Při zvyšování hodnoty $gamma$ se počet špatně označených dopravních značek snižuje, ale roste počet dopravních značek, které nebyly detekovány. Na obrázku 19 je vizualizace rozdílných hodnot parametru $gamma$, kde na levé straně je parametr $gamma$ nastaven na 0.9 a na pravé straně je nastaven na 0.1. Na pravé straně je vidět, že bylo označeno několik objektů jako dopravní značky, přitom se nejedná o značky, ale o část domu, odraz brzdových světel a další.



Obrázek 19: Porovnání detekce za použití různé hodnoty γ . Černo-zelené obdélníky označují objekty, které mají být dopravní značkou.

Z experimentu parametru pro klasifikátor SVM vyplývá, že ovlivnit počet správně nalezených dopravních značek a počet špatně označených dopravních značek nelze moc pomocí primárního klasifikátoru. Oproti tomu u sekundárního klasifikátoru lze ovlivnit počet správně nalezených dopravních značek a počet špatně označených značek. Na základě těchto experimentů jsem zkusil změnit parametr C a γ tak, aby pomocí parametru C bylo nalezeno co nejvíce dopravních značek a pomocí parametru γ byl snížen počet špatně označených dopravních značek. Nastavením parametru C na hodnotu 2 a parametru γ na hodnotu 1.5 byla dosažena úspěšnost 80.27 % s nalezením 604 dopravních značek a špatně bylo označeno 112 značek.

6.2 Experimenty s detektorem

Tato podkapitola se zabývá experimenty s detektorem - ovlivnění přesnosti a času stráveného při detekci dopravních značek na základě změny parametrů detektoru a rozlišení vstupních obrazů. Hlavní klasifikátor má parametr C nastaven na 0.3, sekundární klasifikátor má nastavené parametry na $C = 2.0$ a $\gamma = 1.5$. Počet pozitivních obrazů je 1 213 a negativních 2 000.

6.2.1 Změna parametrů

U detektoru HOG lze změnit parametry, které ovlivní detekci dopravních značek. Jeden z parametrů je o kolik se má detekční okno detektoru zvětšit oproti původní velikosti a druhý parametr je koeficient pro regulaci podobných detekcí v okolí. Rozlišení vstupních snímků je 1 360 x 800 pixelů. Tabulka 5 zobrazuje přesnost detektoru a čas strávený detekcí jednoho snímku v závislosti na změně parametru, který udává, o kolik se má detekční okno detektoru zvětšit v procentech.

Tabulka 5: Změna velikosti detekčního okna a ovlivnění přesnosti a rychlosti detekce.

Zvětšení detekčního okna (%)	Přesnost (%)	TP	FP	FN	Detekce jednoho snímku (<i>ms</i>)
2	80.26	624	142	165	121.96
3	81.05	635	143	154	93.75
4	78.75	141	185	185	82.70
5	80.27	604	112	185	73.22
7	79.84	590	99	199	68.61
10	75.32	528	85	261	63.20

Výsledky ukázaly, že přesnost se moc nemění v závislosti na změně velikosti detekčního okna. Počet detekovaných dopravních značek postupně klesá, ale počet špatně označených dopravních značek taktéž klesá. Oproti tomu rychlost detekce je rychlejší v závislosti na vyšší velikosti detekčního okna.

Druhý parametr je koeficient pro regulaci podobných detekcí v okolí. Tento parametr ovlivňuje, kolik podobných detekcí v okolí se má spojit v jednu detekci. Tabulka 6 zobrazuje přesnost a počet detekcí, kde byla nějaká dopravní značka označena detektorem více jak jednou. Změna velikosti detekčního okna je 5 %.

Tabulka 6: Změna koeficientu pro regulaci podobných detekcí.

Koeficient	Počet opakovaných detekcí	Přesnost (%)	TP	FP	FN
1	4	80.37	608	112	185
2	2	79.41	590	105	201
3	1	77.91	568	100	222
5	0	75.14	526	85	263
7	0	72.30	488	73	301

Z výsledků vyplývá, že nejvyšší úspěšnost byla zaznamenána s koeficientem 1, ale taktéž bylo zaznamenáno nejvíce stejných detekcí, a to 4. Čím vyšší je koeficient, tím je počet opakované detekce nižší, ale stoupá i počet dopravních značek, které nebyly detekovány.

6.2.2 Rozlišení vstupních obrazů

Vstupní obrazy pro testování mají velikost 1 360 x 800 pixelů. Pro experimentování s různým rozlišení vstupních obrazů bylo nutné tyto obrazy zmenšit nebo zvětšit. Taktéž bylo nutné změnit velikost souřadnic dopravních značek uložených v souboru pro detekci. Jsou totiž nastavené na detekci dopravních značek pro snímky o velikosti 1 360 x 800 pixelů. Tabulka 7 zobrazuje rozlišení vstupních obrazů, přesnost při detekci a průměrný čas detekce jednoho snímku. Detekční okno je nastaveno na 5 % a koeficient pro regulaci podobných detekcí je nastaven na 1.

Tabulka 7: Změna koeficientu pro regulaci podobných detekcí.

Rozlišení	Přesnost (%)	TP	FP	FN	Detekce jednoho snímku (<i>ms</i>)
720 x 480	57.47	329	27	460	32.48
1280 x 720	80.48	608	114	181	82.13
1360 x 800	81.05	635	143	154	94.37
1920 x 1080	68.52	556	278	233	185.77
2048 x 1152	66.26	539	299	250	213.02

Výsledky detekce ukázaly, že rozlišení, které je velmi nízké, nedokáže detekovat mnoho dopravních značek. Na druhou stranu je detekováno velmi málo špatně označených značek a rychlost detekce je velmi rychlá. U vysokého rozlišení např. 1 920 x 1 080 pixelů byla rychlost detekce snímků velmi vysoká.

6.3 Výsledky experimentů

Nejlepší výsledky v experimentální části klasifikátoru a detektoru byly dosaženy s parametry primárního klasifikátoru $C = 0.3$ a parametry sekundárního klasifikátoru: $C = 2.0$, $gamma = 1.5$. Koeficient pro regulaci podobných detekcí byl nastaven na 1 a detekční okno bylo nastavené na 3 %. Tabulka 8 zobrazuje kategorii dopravní značky, počet dopravních značek celkově v dané kategorii, počet detekovaných dopravních značek, rozdíl, kolik dopravních značek nebylo nalezeno v dané kategorii a úspěšnost nalezení.

Tabulka 8: Výsledky detekce.

Kategorie značek	Celkový počet	Počet detekovaných značek	Rozdíl	Úspěšnost (%)
Zákazové	421	386	35	91.68
Upravující přednost	155	105	50	67.74
Výstražné	124	70	54	56.45
Příkazové	89	73	16	82.00

Nejvyšší úspěšnost nalezení měla kategorie dopravních značek zákazových, kde byla úspěšnost 92 %. V této kategorii měla většina dopravních značek úspěšnost vyšší než 80 % kromě dopravních značek „zákaz vjezdu všech vozidel“, „konec zákazu předjíždění“ a „konec nejvyšší povolené rychlosti“. Další kategorií jsou příkazové dopravní značky, kde úspěšnost nálezů je 82 %. V této kategorii nebylo moc dopravních značek k testování v trénovacích snímcích (kromě jedné značky, co se vyskytla 52 krát). Kategorie dopravních značek upravující přednost měla úspěšnost 67 %, kde nejmenší úspěšnost nalezení měly dopravní značky „hlavní pozemní komunikace“ 63 % a „dej přednost v jízdě“ 52 %. Nejhuře dopadla kategorie výstražných dopravních značek, kde byla úspěšnost nalezení pouze 56 %. Nejmenší úspěšnost nalezení měly dopravní značky „zúžená vozovka“ a „práce na silnici“.

7 Experimenty na videosekvencích

V této kapitole jsou prezentovány výsledky experimentů na videosekvencích pořízených v reálném prostředí měst Ostrava a Havířov. Na obrázku 20 jsou zobrazeny dopravní značky, které je schopen detektor detekovat.



Obrázek 20: Seznam dopravních značek. [2]

Na disk jsem se rozhodl ukládat pouze detekované objekty z důvodu vysoké náročnosti vytvářet informace o souřadnicích dopravních značek ve videu. Následně jsem uložené detekované objekty na disku porovnával s videem, zda byly všechny dopravní značky detekované. Jelikož detektor dokáže detekovat pouze dopravní značky, které byly naučeny pomocí klasifikátoru SVM, budu brát v potaz pouze dopravní značky, které dokáže detektor rozpoznat a nebudu brát v potaz ostatní dopravní značky. Video se od sebe odlišují tím, že video v Ostravě je pořízeno za slunečného počasí s dobrou viditelností. Jedná se převážně o jízdu po dálnici a výjezd z parkoviště. Video obsahuje celkem 42 značek, které je schopen detektor detekovat. Video v Havířově je pořízeno při zatažené obloze a čelní sklo automobilu bylo špinavé. Video obsahuje průjezd městem, kde se nachází celkem 33 značek, které je schopen detektor detekovat. Tabulka 9 zobrazuje informace o obou videích, kde je uvedena lokalita, celkový počet dopravních značek, počet detekovaných dopravních značek, počet špatně označených značek a celkový počet detekovaných snímků ve videu.

Tabulka 9: Výsledky detekce ve videích.

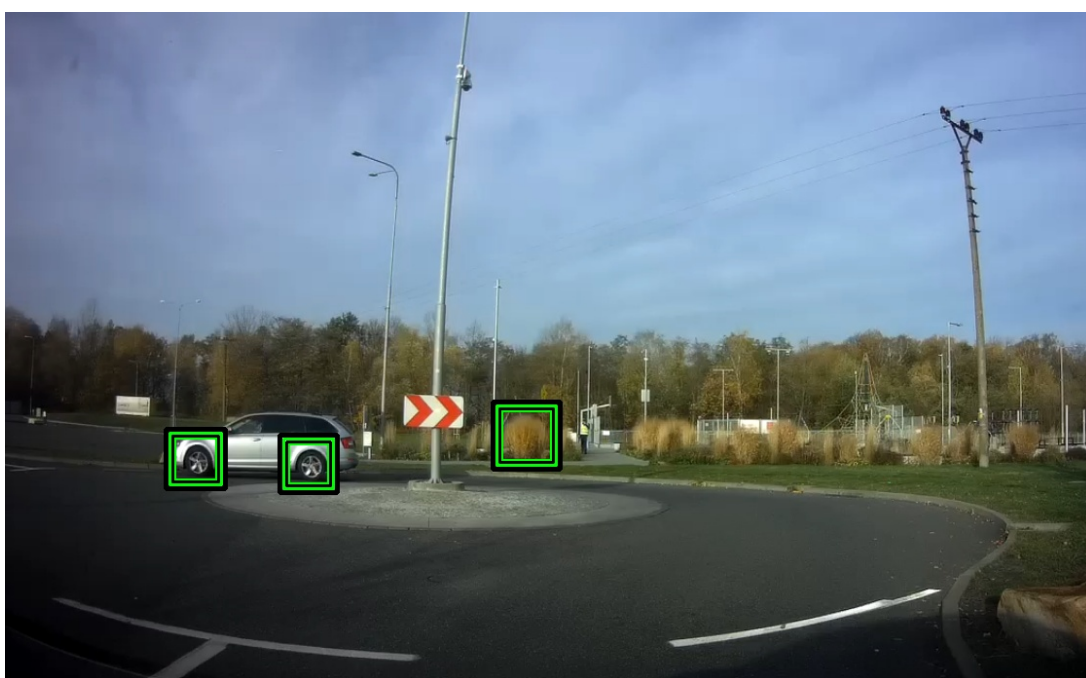
Lokalita	Celkem značek	Detekovaných	Špatně označených	Počet snímků
Ostrava	42	39	18	2 569
Havířov	33	33	16	1 636

U videa v Ostravě nebyly detekovány 3 dopravní značky, především zato můžou světelné podmínky, kdy slunce oslnilo kameru a dopravní značka nebyla detekována (Obr. 21). Bylo špatně označeno celkem 18 objektů, především se jednalo o kola projíždějícího auta (Obr. 22),

dopravní značky, které byly otočené do protisměru a přírodní solitéry. U videa v Havířově bylo nalezeno všech 33 dopravních značek a špatně bylo označeno 16 objektů. Tak jak u videa v Ostravě, i zde byly především označena kola projíždějících aut, otočené dopravní značky do protisměru, světla semaforů a přírodní solitéry.



Obrázek 21: Nedetekovaná dopravní značka. Černo-zelené obdélníky vizualizují nálezy detektoru.



Obrázek 22: Špatně označené objekty. Černo-zelené obdélníky vizualizují nálezy detektoru.

Detektor dokáže detekovat dopravní značky v určité vzdálenosti. Vzdálenost se ale mění podle typu dopravní značky, rychlosti vozidla a světelných podmínek např. počasí. Na obrázku

23 a 24 je porovnání vzdálenosti dvou odlišných dopravních značek v různých podmínkách. Jde vidět, že na obrázku 23 jsou dopravní značky detekovány ve větší vzdálenosti, na obrázku 24 jsou dopravní značky detekovány v menší vzdálenosti.



Obrázek 23: Dopravní značky ve větší vzdálenosti. Černo-zelené obdélníky vizualizují nálezy detektoru.



Obrázek 24: Dopravní značky v krátké vzdálenosti. Černo-zelené obdélníky vizualizují nálezy detektoru.

8 Závěr

Cílem mé bakalářské práce byla detekce dopravních značek v obrazech pomocí obrazových příznaků. Bakalářská práce byla rozdělena na teoretickou část a vlastní implementaci. V teoretické části jsou popsány obecné obrazové příznaky a klasifikátory, které by mohly být v tomto problému aplikovány. Ve vlastní implementaci byla experimentálně ověřena funkčnost, přesnost a rychlost detektoru dopravních značek.

Problémem detekce dopravních značek jsou světelné podmínky, kdy slunce oslní kameru a dopravní značka není detekována. Dalším problémem detekce je detekování dopravních značek určených pro řidiče jedoucí v protisměru nebo označení kol aut a přírodních solitér. Rovněž vzdálenost od vozidla a tvar dopravní značky rozhoduje o její správné detekci. Vysoká rychlost vozidla rovněž snižuje správnost detekce dopravních značek.

Na straně druhé je schopen detektor detekovat i dopravní značky, které nebyly naučeny se rozpoznat pomocí klasifikátoru SVM, ale jsou si podobné tvarem. Úspěšnost detekce dopravních značek ve videosekvencích byla vysoká. Rychlost detekce je závislá na procesoru počítače.

Detekci dopravních značek je možné zlepšit pomocí konvoluční neuronové sítě (CNN). Dnes se používá mnoho různých architektur CNN, které se dají kombinovat i například s klasifikátorem SVM. Jedná se o další cestu k vylepšení detekce dopravních značek.

Literatura

- [1] Zákon č. 361/2000 Sb., o provozu na pozemních komunikacích, § 63.
- [2] HOUBEN, Sebastian, et al. Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark. In: The 2013 international joint conference on neural networks (IJCNN). IEEE, 2013. p. 1-8.
- [3] TIMOFTE, Radu; ZIMMERMANN, Karel; VAN GOOL, Luc. Multi-view traffic sign detection, recognition, and 3D localisation. Machine vision and applications, 2014, 25.3: 633-647.
- [4] SHAKHURO, Vladislav Igorevich; KONOUCHE, A. S. Russian traffic sign images dataset. Computer Optics, 2016, 40.2: 294-300.
- [5] LARSSON, Fredrik; FELSBURG, Michael. Using Fourier descriptors and spatial models for traffic sign recognition. In: Scandinavian conference on image analysis. Springer, Berlin, Heidelberg, 2011. p. 238-249.
- [6] MOGELMOSE, Andreas; TRIVEDI, Mohan Manubhai; MOESLUND, Thomas B. Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey. IEEE Transactions on Intelligent Transportation Systems, 2012, 13.4: 1484-1497.
- [7] ELLAHYANI, Ayoub, et al. Traffic sign detection and recognition using features combination and random forests. International Journal of Advanced Computer Science and Applications, 2016, 7.1: 683-693.
- [8] MEENA, K.; SURULIANDI, A. Local binary patterns and its variants for face recognition. In: 2011 International Conference on Recent Trends in Information Technology (ICRTIT). IEEE, 2011. p. 782-786.
- [9] VIOLA, Paul, et al. Rapid object detection using a boosted cascade of simple features. CVPR (1), 2001, 1: 511-518.
- [10] HRŮZ, Marek. LBP, HoG [online]. Plzeň, 2015. Dostupné z: <http://www.kky.zcu.cz/uploads/courses/mpv/05/materialy05.pdf>
- [11] Support vector machines (SVM). [online]. Dostupné z: https://is.muni.cz/el/1433/podzim2006/PA034/09_SVM.pdf
- [12] [online]. Dostupné z: <https://docs.opencv.org/>
- [13] ZHU, Zhe, et al. Traffic-sign detection and classification in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. p. 2110-2118.

- [14] OVERETT, Gary; PETERSSON, Lars. Large scale sign detection using HOG feature variants. In: 2011 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2011. p. 326-331.
- [15] MAŠEK, Jan. Detekce objektů v obraze s pomocí Haarových příznaků: diplomová práce [online]. Brno: VUT, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2012. 72 s. Vedoucí práce byl Ing. Radim Burget, Ph.D. Dostupné z: http://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=51400
- [16] MALACH, Tobiáš; BAMBUCH, Petr; MALACH, Jindřich. Detekce obličejů v obraze s využitím prostředí MATLAB®. Proceedings of the 19 Technical Computing Prague. Praha (Czech Republic), 2011, 2011, 78.
- [17] BRKIC, Karla. An overview of traffic sign detection methods. Department of Electronics, Microelectronics, Computer and Intelligent Systems Faculty of Electrical Engineering and Computing Unska, 2010, 3: 10000.
- [18] BERKAYA, Selcan Kaplan, et al. On circular traffic sign detection and recognition. Expert Systems with Applications, 2016, 48: 67-75.
- [19] KUANG, Xianyan; FU, Wenbin; YANG, Liu. Real-Time Detection and Recognition of Road Traffic Signs using MSER and Random Forests. International Journal of Online Engineering, 2018, 14.3.
- [20] EL MARGAE, Samira, et al. Multi-stage fusion of local and global features based classification for traffic sign recognition. In: 2014 International Conference on Multimedia Computing and Systems (ICMCS). IEEE, 2014. p. 495-499.
- [21] Chirakkal, Vinjohn & Han, Dong. (2015). Traffic Sign Recognition based on MB-LBP and HOG Descriptors. 10.13140/RG.2.2.19657.65124.
- [22] [online]. Dostupné z: <https://opencv.org/>
- [23] AHONEN, Timo; HADID, Abdenour; PIETIKAINEN, Matti. Face description with local binary patterns: Application to face recognition. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2006, 12: 2037-2041.

A Příloha v IS EDISON

obsah přílohy

- Zdrojové kódy programu pro detekci dopravních značek
- Návod pro spuštění programu